Claims

What is claimed is:

1. A method of transmitting data frames from a sending unit to a receiving unit in a data transmission network comprising at least a backbone wherein the data are transmitted over high speed links enabling long Maximum Transmission Units (MTU) between an ingress node connected to the sending unit by a first access link and an egress node connected to the receiving node by a second access link, with at least one of the first and second access links being a low speed access link requiring the data frames to be segmented into short MTUs between the sending unit and the ingress node and between the egress node and the receiving unit , the method comprising the steps of

assembling, in the ingress node, a plurality of consecutive segmented data frames belonging to the same flow of data transmitted from the sending unit to the ingress node into an assembled data frame corresponding to one of the long MTUs,

transmitting, the assembled data frame over the backbone from the ingress node to the egress node at a high speed authorized by the backbone links,

de-assembling, the assembled data frame, in the egress node into consecutive segmented data frames corresponding to the short MTUs, and,

transmitting, the consecutive segmented data frames from the egress node to the receiving unit.

2. A method according to claim 1, wherein the assembling step comprises the steps of

defining in the assembled data frame a plurality of assembled parts each of them comprising a data field containing the data of the corresponding frame of the plurality of consecutive segmented data

frames and an assembly header containing at least the length of the assembled part, and

defining a main protocol header preceding the plurality of assembled parts and containing the same protocol information as the protocol headers of the segmented data frames and new information relating to the assembled data frame.

3. A method according to claim 2, wherein the assembling step comprises the step of building the assembly header of each assembled part as a differential header, with respect to the main protocol header, including one position field giving the position of the first byte different from the main protocol header, one length field giving the number of consecutive bytes different from the main protocol header and all the consecutive different bytes identified by the position and length fields.

4. A method according to claim 2, wherein the assembling step comprises the step of gathering as many consecutive data frames belonging to the same flow as necessary to have a total size of all these frames equal to or just below a predetermined limit corresponding to the one of the long MTUs.

5. A method according to claim 3, wherein the assembling step comprises the step of gathering as many consecutive data frames belonging to the same flow as necessary to have a total size of all these frames equal to or just below a predetermined limit corresponding to the one of the long MTUs.

6. A method according to claim 5, wherein the step of assembling a plurality of consecutive data frames belonging to the same flow of data includes the step of looking up a table stored in the ingress node to check whether there is an entry in the table corresponding to the flow of data.

7. A method according to claim 6,wherein the assembling step comprises the step of creating the entry in the table when the first frame of the flow of data is received by the ingress node.

8. A method according to claim 7,wherein the assembling step comprises the step of storing the plurality of consecutive data frames belonging to the same flow of data in a frame buffer allocated in the ingress node, before being assembled into the assembled frame when the total size reaches the predetermined limit.

9. A method according to claim 8,wherein the assembling step comprises the step of storing each one of the plurality of frames in a location of the frame buffer which is defined by a « next buffer » field in the location of the buffer in which is stored the preceding frame of the plurality of frames.

10. A method according to claim 9 wherein the assembling step comprises the step of defining in each location in the frame buffer a field containing the size of the data of the corresponding frame.

11. A method according to claim 10,wherein the assembling step comprises the step of defining in the entry in the table defining the flow of data an « origin » field pointing to the address of the location wherein is stored the first frame of the plurality of frames stored in the frame buffer and a « buffer address » field pointing to the address of the location in the frame buffer in which the current frame is to be stored.

12. A method according to claim 11,wherein the assembling step further comprises the step of defining in the entry in the table a « timer » field defining a time counter which is set at a predetermined value when the first frame of the plurality of frames is received by the ingress node and which is regularly decremented until zero, the step of assembling the assembled frame being performed if the predetermined value is reached before the total size has a value equal to or just below the predetermined limit.

13. A method according to claim 12, wherein the assembling step further the step of periodically decrementing the « timer » fields in all entries of the table and checking whether the value of each decremented « timer » field is zero.

14. A method according to claim 1, further comprising the step of identifying the protocol of each frame received by the ingress node and performing the step of assembling the consecutive segmented frames belonging to the same flow of data only if the protocol of the received frame is identified.

15. A method according to claim 13, further comprising the step of identifying the protocol of each frame received by the ingress node and performing the step of assembling the consecutive segmented frames belonging to the same flow of data only if the protocol of the received frame is identified.

16. A method according to claim 1, wherein the step of de-assembling the assembled data frame in the egress node consists for each assembled part of the assembled frame, in building the frame header of each segmented frame by using the main protocol header and the differential header of the assembled part the data of the segmented frame being the data of the assembled part.

17. A method according to claim 15, wherein the step of de-assembling the assembled data frame in the egress node consists for each assembled part of the assembled frame, in building the frame header of each segmented frame by using the main protocol header and the differential header of the assembled part the data of the segmented frame being the data of the assembled part.

18. A data transmission system comprising :

at least a backbone comprising an ingress node and an egress node, the backbone allowing high speed transmission of data and enabling long Maximum Transmission Units (MTU) between ingress node and egress node,

a sending unit connected to the ingress node via a first access link and a receiving unit connected to the egress node via a second access link, the first and second access links being low speed access links requiring the data frames to be segmented into short MTUs between the sending unit and the ingress node and between the egress node and the receiving unit,

assembling instrumentalities located in the ingress node to assemble a plurality of consecutive segmented data frames belonging to the same flow of data transmitted from the sending unit to the ingress node into an assembled data frame corresponding to one of the long MTUs.

19. A data transmission system according to claim 18, wherein the assembling instrumentalities comprise

a storage unit storing a table wherein each entry is allocated to each flow of data and contains information on the flow of data, a frame buffer for storing each segmented frame of the plurality of frames, a list of pointers pointing to the free locations of the frame buffers and a FIFO for storing consecutive processed frame data for building the assembled frame data,

a lookup unit for looking up the table to find the entry corresponding to a given frame or creating a new entry if necessary,

a frame processing unit receiving the plurality of consecutive segmented data frames and, for each frame, requesting its corresponding data flow information to the lookup unit and storing frame data in the frame buffer using the list of pointers to find a free frame buffer entry, and

an assembly processing unit reading all the data frames stored in the frame buffer by the frame processing for a given data flow, processing them, storing them in the FIFO and reading them to build and send the assembled data frame.

20. A data transmission system according to claim 19, wherein an entry in the table includes an « origin » field pointing to the address of the location wherein is stored the first frame of the plurality of frames stored in the frame buffer and a « buffer address » field pointing to the address of the location in the frame buffer in which the current frame is to be stored.

21. A data transmission system according to claim 20, wherein the entry in the table includes a « total

size » field storing the total size of the consecutive segmented frames of the same flow, the segmented frames being processed into the assembled data by the assembly processing unit associated with the FIFO as soon as the value of the « total size » field is equal or just below a predetermined limit.

22. A data transmission system according to claim 21, wherein the entry in the table further includes a « timer » field defining a time counter which is set to a predetermined value when the first frame of the plurality of frames is received by the ingress node and which is regularly decremented until zero, the step of assembling the assembled frame being performed if the predetermined value is reached before the total size has a value equal to or just below the predetermined limit.

23. A data transmission system according to claim 22, wherein each location of the frame buffer includes, further to the header and the data of the frame, a « next buffer » field defining the location in the frame buffer in which the next frame is to be stored and a « data size » field containing the size of the data of the stored frame.

24. A data transmission system according to claim 23, wherein the assembly processing unit builds a main protocol header for the assembled data frame and a plurality of assembled parts, each assembled part including a differential header with respect to the main protocol header and the data of the corresponding segmented frame, the assembled parts being serially stored in the FIFO and the assembled parts preceded by the main protocol header being serially transmitted over the backbone when the last assembled part of the assembled frame has been stored in the FIFO.

25. A data transmission system according to claim 24, further comprising protocol processing instrumentalities for identifying the protocol of the frame received by the ingress node and processing the frame if necessary before performing the assembly of the frame only if the protocol of the frame is identified.

26. A data transmission system according to claim 25, wherein the egress node comprises

de-assembling instrumentalities comprising assembled frame processing for identifying the protocol of the frame received by the egress node, a header processing unit for building the protocol header of each segmented data frame to be transmitted from the egress node to the receiving unit, a data handling unit for segmenting the data of the assembled data frame into the data fields of the segmented frames and a frame buffer for storing each re-built segmented frame before transmitting it to the receiving unit.

27. A data transmission system according to claim 19, wherein the assembly processing unit comprises building instrumentalities to build a main protocol header for the assembled data frame and a plurality of assembled parts, each assembled part including a differential header with respect to the main protocol header and the data of the corresponding segmented frame, the assembled parts being serially stored in the FIFO and the assembled parts preceded by the main protocol header being serially transmitted over the backbone when the last assembled part of the assembled frame has been stored in the FIFO.

28. A data transmission system according to claim 26, wherein the assembly processing unit comprises building instrumentalities to build a main protocol header for the assembled data frame and a plurality of assembled parts, each assembled part including a differential header with respect to the main protocol header and the data of the corresponding segmented frame, the assembled parts being serially stored in the FIFO and the assembled parts preceded by the main protocol header being serially transmitted over the backbone when the last assembled part of the assembled frame has been stored in the FIFO.

29. A data transmission system according to claim 18, further comprising a protocol processing instrumentalities for identifying the protocol of the frame received by the ingress node and processing the frame if necessary before performing the assembly of the frame only if the protocol of the frame is identified.

30. A data transmission system according to claim 18, wherein the egress node comprises de-assembling instrumentalities comprising assembled frame processing for identifying the protocol of the frame received by the egress node, a header processing unit for building the protocol header of each segmented data frame to be transmitted from the egress node to the receiving unit, a data handling unit for segmenting the data of the assembled data frame into the data fields of the segmented frames and a frame buffer for storing each re-built segmented frame before transmitting it to the receiving unit.